

# Video Adaptation for Mobile Digital Television

R. Garrido-Cantos\*, J. De Cock\*\*, J.L. Martínez\*, S. Van Leuven\*\*, P. Cuenca\*, A. Garrido\*, and R. Van de Walle\*\*

\* Albacete Research Institute of Informatics, University of Castilla-La Mancha, Albacete, Spain

\*\* Department of Electronics and Information Systems - Multimedia Lab, Ghent University - IBBT, Ghent, Belgium  
{charo, joseluismm, pcuenca, antonio}@dsi.uclm.es, {jan.decock, sebastiaan.vanleuven, rik.vandewalle}@ugent.be

**Abstract**—Mobile digital television is one of the new services introduced recently by telecommunications operators in the market. Due to the possibilities of personalization and interaction provided, together with the increasing demand of this type of portable services, it would be expected to be a successful technology in near future. Video contents stored and transmitted over the networks deployed to provide mobile digital television need to be compressed to reduce the resources required. The compression scheme chosen by the great majority of these networks is H.264/AVC. Compressed video bitstreams have to be adapted to heterogeneous networks and a wide range of terminals. To deal with this problem scalable video coding schemes were proposed and standardized providing temporal, spatial and quality scalability using layers within the encoded bitstream. Because existing H.264/AVC contents cannot benefit from scalability tools, efficient techniques for migration of single-layer to scalable contents are desirable for supporting these mobile digital television systems. This paper proposes a technique to convert from single-layer H.264/AVC bitstream to a scalable bitstream with temporal scalability. Applying this approach, a reduction of 60% of coding complexity is achieved while maintaining the coding efficiency.

## I. INTRODUCTION

In the last few years, mobile consumers demand for content is growing. Mobile services have been gaining popularity and the possibility of receiving digital Television (TV) everywhere makes of *Mobile Digital TV* (MDTV) a successful technology in the near future. Several factors are essential for MDTV to have commercial impact: handheld devices with suitable displays, appropriate multimedia compression technologies and adequate transport systems.

In order to transmit MDTV, some new network technologies have been specifically deployed to overcome the difficulties that arise with these types of environments and terminals. These network technologies have been designed to deal with the special issues that appear with handheld devices such as battery lifetime, computing capacity or screen size and the special requirements regarding mobile reception such as handover, bandwidth or indoor reception.

The newest one, *Advanced Television Systems Committee - Mobile/Handheld* (ATSC-M/H) [1] has been standardized recently. Other network technologies are *Digital Video Broadcasting Handheld* (DVB-H) [2] or *Multimedia Broadcast/Multicast Service* (MBMS) [3]. The first two ones are extensions of their terrestrial equivalent *Digital Video Broadcast Terrestrial* (DVB-T) [4] and *Advanced Television Systems Committee A/53* (ATSC A/53) [5], respectively, and the last one is built

over the existing 3G network. All of them use broadcasting to deliver unidirectional and real-time media bitstreams, although MBMS is able to use multicast transmission. At the same time, these network technologies can provide interactivity using mobile phone network (see Fig. 1).

On the other hand, reliable reception of video contents by the mobile devices poses additional constraints because of the dynamic nature of the links and the limited resources of the mobile reception devices. Therefore, real time video adaptation for mobile devices will play a crucial role in the future mobile digital television. The compressed video bitstream will have to be adapted to the network connections and different characteristics of devices to ensure high quality image continuously.

For this reason, *Scalable Video Coding* (SVC) schemes have gained popularity in the last years. The main idea of SVC is to encode the video as one base layer and a few enhancement layers, so that lower bitrates, spatial and temporal resolutions could be obtained by truncating certain layers from the original bitstream to adapt to the communication channel bandwidth and/or user device capabilities. Recently, *Moving Picture Experts Group* (MPEG) and *Video Coding Experts Group* (VCEG) have standardized a new scalable extension of H.264/AVC that is denoted as SVC [6]. SVC makes possible to encode scalable video bitstreams providing different types of scalability such as temporal, spatial and *Signal to Noise Ratio* (SNR) in a flexible manner. Temporal scalability in SVC is provided by using *Hierarchical prediction structures*, spatial scalability is achieved by encoding each spatial resolution into one layer and quality-SNR scalability is intended to give different levels of detail and fidelity to the original video. To remove redundancy between layers inter-layer prediction mechanisms are applied.

In this way, ATSC-M/H and DVB-H systems have established recently a set of video coding specifications where H.264/AVC and SVC are chosen to transmit video in these networks and they also have defined the RTP packetization for video elementary streams [7][8].

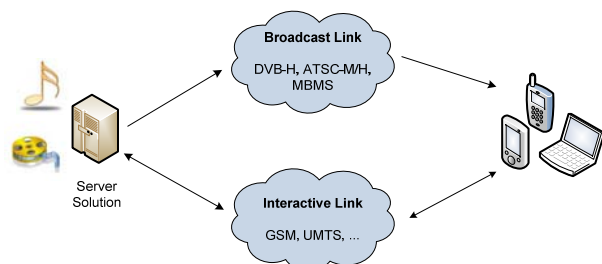


Figure 1. Broadcast mobile TV network with interactivity

Despite these scalability tools, most of the video contents today are still created in a single-layer format (H.264/AVC video streams), so it becomes necessary to develop alternative techniques to enable video adaption. In this paper, video transcoding [9] is proposed for enabling efficient adaption of H.264/AVC to SVC video streams. Its efficiency is obtained by reusing as much information as possible from the original bitstream, such as motion information. The ultimate goal is to perform the required adaptation process faster than the straightforward concatenation (cascade) of decoder and encoder. In particular, this paper describes a technique for transcoding from a single-layer H.264/AVC bitstream without temporal scalability (typically IBBP GOP pattern) to an SVC bitstream with temporal scalability with hierarchical B prediction structures that it is capable to reduce coding complexity around to 60% while maintaining coding efficiency.

The remainder of this paper is organized as follows. In Section II, the state-of-the-art for H.264/AVC to SVC transcoding is discussed. Section III describes the temporal scalability technique in SVC. In Section IV our approach is described. In Section V the implementation results are shown. Finally, in Section VI conclusions are presented.

## II. RELATED WORK

Since it is beneficial for broadcasters and content distributors to have scalable bitstreams at their disposal, efficient techniques for migration of H.264/AVC to a SVC format are desirable. Due to its computational efficiency, transcoding can be used for introducing scalability in compressed, single-layer bitstreams. In this way, re-encoding can be avoided when migrating legacy content to a scalable format. A number of techniques have been proposed in the past for introducing scalability in compressed bitstreams. The majority of the proposals are related to quality-SNR scalability, although there are some related to spatial and temporal scalability.

Respecting quality-SNR scalability, a technique was studied for transcoding from hierarchically encoded H.264/AVC to *Fine-Grain Scalability* (FGS) streams in [10]. Although it was the first work in this type of transcoding, does not have a great relevance since this technique for providing quality-SNR scalability was removed from the following versions of the standard due to its high computational complexity. In [11], different architectures for transcoding from single layer H.264/AVC bitstream to SNR scalable SVC streams with *Coarse-Grain Scalability* (CGS) layers were proposed that depends on the macroblock type. Moreover, the normative bitstream rewriting process implemented in SVC standard to convert SVC to H.264/AVC bitstream is used to reduce the computational complexity of the proposed architectures.

For spatial scalability, a proposal was presented in [12]. It presented an algorithm for converting a single layer H.264/AVC bitstream to a multi-layer spatially scalable SVC video bitstream, containing layers of video with different spatial resolution. Using a full-decode full-encode algorithm as starting point, some modification are made to reuse information available after decoding a H.264/AVC bitstream for motion estimation and refinement processes on the encoder. The scalability is achieved by an information downscaling algorithm which

use the top enhancement layer (this layer has the same resolution as the original video output) to produce different spatial layers of the output SVC bitstream.

For temporal scalability, a transcoding method from H.264/AVC P-picture based bitstream to a SVC bitstream was presented in [13]. In this approach, the H.264/AVC bitstream is transcoded to a two layers of P-pictures (one with reference pictures and another with non-reference ones). Then, this bitstream is transformed to a SVC bitstream by syntax adaptation.

## III. TEMPORAL SCALABILITY IN SVC

A bitstream provides temporal scalability when can be divided into a temporal base layer and one or more temporal enhancement layers, so that if all the enhancement temporal layers with an identifier greater than one specific temporal layer are removed, the remaining temporal layers forms another valid bitstream for the decoder.

In H.264/AVC and for extension in SVC, any picture can be marked as reference picture and used for motion compensated prediction of following pictures. This feature allows the coding of picture sequences with arbitrary temporal dependencies. Hence, for supporting temporal scalability with a reasonable number of temporal layers, no changes to the design of H.264/AVC were required. In this way, to achieve temporal scalability, SVC links its reference and predicted frames using *hierarchical prediction structures* [14] which define the temporal layering of the final structure.

With *hierarchical prediction structures*, key pictures (typically I or P frames) are coded in regular intervals by using only previous key pictures as references. The pictures between two key pictures are hierarchically predicted and together with the succeeding key picture are known as *Group of Pictures* (GOP). The sequence of key pictures represents the lowest temporal (temporal base layer) which can be increase with the non key pictures that are divided into enhancement layers.

There are different structures for enabling temporal scalability, but the typical GOP structure is based on hierarchical B pictures, which is also used by default in the JSVM reference encoder software [15]. The number of temporal layers is thus equal to (1). One of these structures, with dyadic structure, GOP of 8 (17BP pattern) and therefore four temporal layers, is illustrated in Fig. 2.

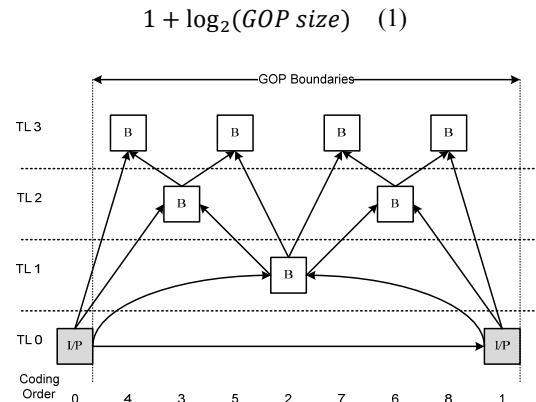


Figure 2. Hierarchical B prediction structure with four temporal layers (TL)

#### IV. H.264/AVC-TO-SVC TRANSCODING

One of the most time consuming tasks carried out at H.264/AVC and SVC encoders is the *Motion Estimation* (ME). The idea behind the proposed transcoder consists of reusing the motion information that can be gathered in the H.264/AVC decoding algorithm (as part of the transcoder) to accelerate the SVC encoding process (also included in the transcoder). In this framework, the proposed transcoder tackles the ME complexity reduction by reusing the *Motion Vectors* (MVs) calculated in H.264/AVC in order to define smaller search areas in SVC.

##### A. Motivation

The idea of ME consists of eliminating the temporal redundancy in a way to determine the movement of the scene. For this purpose, in H.264/AVC MVs between every *Macroblock* (MB) or sub-MB and the block which generates the lowest residual inside the search area of the reference frame are calculated. These MVs represent, approximately, the amount of movement of the MB.

Since the MVs, generated by H.264/AVC and transmitted into the encoded bitstream, represent, approximately, the amount of movement of the frame, they can be reused to accelerate the SVC motion estimation process by reducing the search area dynamically and efficiently.

The main challenge to overcome in this transcoding architecture is the mismatching between GOP sizes, GOP patterns and prediction structures. While the starting encoded bitstream in H.264/AVC is formed by IBBP GOP patterns without temporal scalability, the final SVC bitstream needs conforming hierarchical structures (see Fig. 2). This fact leads to different MVs in both H.264/AVC and SVC. Furthermore, MB partitions developed by H.264/AVC can be different from SVC ones as shown in Fig. 3 so the number of MVs associated to an H.264/AVC MB can be different from the number of MVs associated to the corresponding SVC MB as illustrated in solid line in Fig. 4.

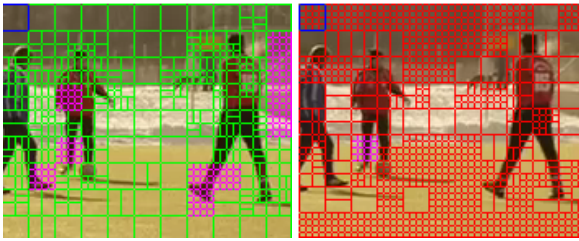


Figure 3. MB partitions generated by H.264/AVC and SVC for the 4<sup>th</sup> frame in Soccer QCIF sequence

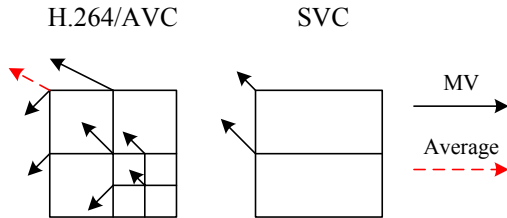


Figure 4. Example MB in H.264/AVC with its MVs and the matching MB in SVC with its corresponding MVs

As Fig. 4 shows, there is not always a one-to-one mapping between previously calculated H.264/AVC MVs and the incoming SVC MVs. The present approach tries to tackle with this problem.

##### B. First stage: Initial Dynamic Motion Window

This paper proposes a *Dynamic Motion Window* (DMW) technique that uses the incoming MVs from H.264/AVC to determine a small area to find the real MVs calculated in SVC which is depicted in Fig. 5. This smaller search area is determined by the circumference centered in (0,0) point for each MB or sub-MB. This circumference has a radius which varies dynamically depending on the length of the average of the incoming vector for a specific MB (in dash line in Fig. 4) and the temporal layer which the frame is in. The average of the incoming MVs of a determined MB is used to overcome the situation explained previously where the number of MVs associated to a MB are different. The dependency of the layer will be explained in Section IV.C.

So in this way, the initial search window is limited by the area  $S$  defined in (2).

$$S = \{(x, y) | (x, y) \in (A \cap C)\} \quad (2)$$

Where  $(x, y)$  are the coordinates to check,  $A$  is the search range used by SVC and  $C$  is the circumference which restricts the initial search area with centre on the upper left corner of the MB or sub-MB.  $C$  is defined by (3).

$$C^2 = r_x^2 + r_y^2 \quad (3)$$

Where  $r_x$  and  $r_y$  are calculated from (4) and (5) depending of the average of MVs of the H.264/AVC MB ( $MV_x$  and  $MV_y$ ) or a minimum value of 1 to avoid applying too small search ranges.

$$r_x = \max(MV_x, 1) \quad (4)$$

$$r_y = \max(MV_y, 1) \quad (5)$$

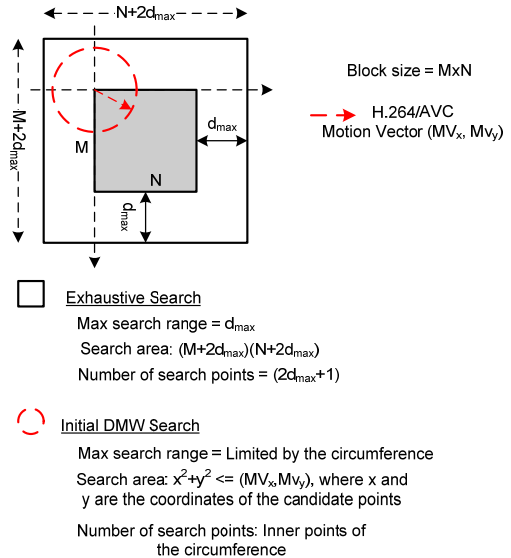


Figure 5. Proposed initial dynamic motion window

Both H.264/AVC and SVC use two lists of previously-coded reference frames (*list0* and *list1*), before or after the current picture in temporal order in B pictures (bidirectional) for prediction. For P pictures only *list0* is used.

Due the different GOP patterns between H.264/AVC and SVC, it is usual to have cases where MVs extracted from H.264/AVC are obtained with a reference of a *list0*, but SVC needs the reference from the *list1* or vice versa or even a bidirectional prediction is done requiring MVs of both lists. In these cases, the supposition is made that the length of the MV of both lists for a MB is the same.

### C. Second Stage: Adjusting Initial DMW Length

As it mentioned previously, MVs generated in H.264/AVC are re-used to generate a new small area defined by a circumference with the incoming MV for this MB as its radius.

Something to keep in mind is that these MVs for each MB have been calculated in H.264/AVC using a reference frame that could have a different distance from the current frame. In general, GOP structures in SVC with temporal scalability lead to longer distances between a frame and its reference frame than in H.264/AVC. As it could be seen in Fig. 2, with hierarchical pictures structures, the distance between both frames is longer when the temporal layer decreases.

To deal with this different prediction distance, a correction factor is introduced so the circumference generated previously is multiplied by a factor that depends on which temporal layer the current frame is in. This process is illustrated in Fig. 6.

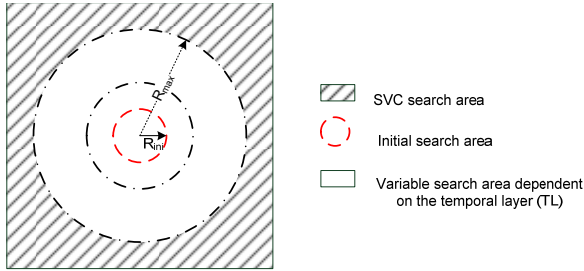


Figure 6. Variation of initial search area depending on temporal layer

Therefore, (4) and (5) are multiplied by this correction factor, so  $r_x$  and  $r_y$  will be calculated using (6) and (7).

$$r_x = coef(n) \cdot \max(MV_x, 1) \quad (6)$$

$$r_y = coef(n) \cdot \max(MV_y, 1) \quad (7)$$

Here,  $coef$  depends on the number of the temporal layer ( $n$ ) where the frame is in as defined in (8).

$$coef(n) = \frac{GOP\ length}{2^n} \quad (8)$$

## V. IMPLEMENTATION RESULTS

In this section, results from the implementation of the proposal described in previous section are shown. Test sequences with varying characteristics were used, namely

Hall, City, Foreman, Soccer, Harbour, Crew, Football and Mobile in CIF resolution (30 Hz) and QCIF resolution (15 Hz).

These sequences were encoded using the H.264/AVC *Joint Model* (JM) reference software, version 16.2 [16], with an IBBP pattern with a fixed QP = 28 in a trade-off between quality and bitrate. Then, for reference results, encoded bitstreams are decoded and re-encoded using the JSVM software, version 9.19.3 [15] with temporal scalability and different values of QP (28, 32, 36, 40). For results of the proposal, encoded bitstreams in H.264/AVC are transcoded using the technique described in Section IV. A typical GOP length of 16 is used for CIF sequences and a GOP length of 8 for QCIF sequences, which corresponds to inserting a key picture roughly every 0.5s.

In SVC encoding, most of the time is spent on the higher temporal enhancement layers. Table I and Table II show the percent of time distribution of the encoding time per layer. As is shown, around 80% (TL2 and TL3 in Table I and TL3 and TL4 in Table II) of the time is spent for these layers. Therefore, the proposed DMW algorithm depicted in Section IV has been applied on the upper temporal layers (last two layers in QCIF and CIF). The remaining temporal layers will be decoded and re-encoded completely.

Table III and Table IV show  $\Delta PSNR$ ,  $\Delta Bitrate$  and  $\Delta Time$  when our technique is applied compared to the more complex reference transcoder.  $\Delta Time$  is calculated for the full sequence and for the last two temporal layers where the approach is applied.  $\Delta PSNR$  and  $\Delta Bitrate$  are calculated according to the specified in [17]. For PSNR, the averaged PSNR values of luminance (Y) and chrominance (U, V) are used. This averaged-global PSNR is based on (9).

$$\overline{PSNR} = \frac{4 \cdot PSNR_Y + PSNR_U + PSNR_V}{6} \quad (9)$$

In order to evaluate the time-saving of the proposal, (10) is calculated where  $T_{ref}$  denotes the coding time used by the SVC reference software encoder and  $T_{pro}$  is the time spent by the proposed algorithm. In Tables III and IV, for  $\Delta T_{total}$  these times are calculated over the entire sequence, whereas for  $\Delta T_{partial}$  only the time spent in the last two temporal layers where the proposal is applied is taken into account.

$$\Delta Time (\%) = \frac{T_{pro} - T_{ref}}{T_{ref}} \cdot 100 \quad (10)$$

The values obtained with the proposed transcoder are very close to the results obtained when applying the reference transcoder (re-encoder): the average PSNR lost over the reference is 0.05 dB, with an average increase of bitrate around 1.4% in QCIF and 2.5% in CIF resolution and achieving around 47.5% of reduction of computational complexity in the full sequence and 60% in the specific layers.

The resulting *Rate-Distortion* (RD) curves for the SVC bitstreams are shown in Fig. 7 and Fig. 8 where it can be seen that our proposal for transcoding is able to approach the RD-optimal transcoded (re-encoded) reference without any significantly loss.

TABLE I.

ENCODING TIME FOR EACH TEMPORAL LAYER (TL) USING QCIF

Sequence	Encoding time (%) of every temporal layer GOP = 8 – QCIF (15 Hz)			
	TL0	TL1	TL2	TL3
Hall	5.08	13.52	27.10	54.30
City	5.10	13.51	27.02	54.37
Foreman	4.71	13.73	27.32	54.24
Soccer	4.99	13.44	27.13	54.44
Harbour	5.13	13.54	27.00	54.33
Crew	5.08	13.39	27.02	54.51
Football	4.67	13.73	27.26	54.34
Mobile	4.72	13.65	27.37	54.26
Average	4.94	13.56	27.15	54.35

TABLE II.

ENCODING TIME FOR EACH TEMPORAL LAYER (TL) USING CIF

Sequence	Encoding time (%) of every temporal layer GOP = 16 – CIF (30 Hz)				
	TL0	TL1	TL 2	TL 3	TL4
Hall	1.57	6.56	13.15	26.39	52.33
City	2.43	6.46	12.92	26.01	52.18
Foreman	1.52	6.63	13.11	26.34	52.40
Soccer	1.54	6.55	13.11	26.35	52.45
Harbour	2.44	6.46	12.92	26.01	52.17
Crew	2.42	6.41	12.90	26.01	52.26
Football	1.55	6.58	13.13	26.34	52.40
Mobile	1.49	6.61	13.09	26.32	52.50
Average	1.87	6.53	13.04	26.22	52.34

TABLE III.

RD PERFORMANCE AND TIME SAVINGS OF THE APPROACH FOR QCIF RESOLUTION

RD performance and time savings of AVC/SVC transcoder GOP = 8 - QCIF (15 Hz)				
Sequence	APSNR (dB)	ΔBitrate (%)	ΔTime (%)	
			Full Seq.	Partial
Hall	0.022	0.62	-72.27	-89.35
City	-0.028	1.66	-46.74	-60.40
Foreman	-0.010	0.92	-41.83	-51.83
Soccer	-0.123	4.13	-34.19	-41.39
Harbour	0.005	0.32	-70.25	-86.32
Crew	-0.026	1.29	-27.09	-33.77
Football	-0.029	1.17	-19.08	-23.71
Mobile	-0.018	0.76	-69.68	-86.13
Average	-0.026	1.36	-47.64	-59.11

TABLE IV.

RD PERFORMANCE AND TIME SAVINGS OF THE APPROACH FOR CIF RESOLUTION

RD performance and time savings of AVC/SVC transcoder GOP = 16 - CIF (30 Hz)				
Sequence	APSNR (dB)	ΔBitrate (%)	ΔTime (%)	
			Full Seq.	Partial
Hall	0.003	0.64	-66.01	-85.43
City	-0.100	2.61	-52.22	-67.85
Foreman	-0.035	1.30	-44.99	-58.56
Soccer	-0.121	5.76	-34.89	-45.59
Harbour	0.011	0.31	-64.40	-83.43
Crew	-0.043	2.08	-31.50	-41.28
Football	-0.062	3.17	-19.89	-27.67
Mobile	-0.016	0.79	-66.48	-86.45
Average	-0.045	2.08	-47.55	-62.03

## VI. CONCLUSIONS

This work presents an approach for H.264/AVC to SVC transcoding with temporal scalability. By reusing information available after decoding the H.264/AVC bitstream, the ME process for the higher temporal layers in SVC can be accelerated by building a DMW with incoming motion vectors as the radius and applying a correction coefficient depending on the temporal layer where the frame is in. Experimental results applying this approach show that it is capable to reduce the coding complexity by around 60% while maintaining the coding efficiency.

## ACKNOWLEDGMENTS

This work was supported by the Spanish MEC and MICINN, as well as European Commission FEDER funds, under grants CSD2006-00046, TIN2009-14475-C04, and it was also partly supported by JCCM funds under grants PEI09-0037-2328 and PII2I09-0045-9916.

## REFERENCES

- [1] Advanced Television System Committee: ATSC-Mobile DTV Standard, A/153 ATSC Mobile Digital Television System. October 2009.
- [2] European Broadcasting Union: ETSI TR 102 377 V1.4.1: Digital Video Broadcasting (DVB); DVB-H Implementation Guidelines. June 2009.
- [3] 3GPP: TS 23.246 V9.4: Multimedia Broadcast/ Multicast Service (MBMS); Architecture and functional description. March 2010.
- [4] European Broadcasting Union: ETSI TR 101 190 V1.3.1: Digital Video Broadcasting (DVB); Implementation guidelines for DVB terrestrial services; Transmission aspects. October 2008.
- [5] Advanced Television System Committee: ATSC Digital Television Standard: Part 1 – Digital Television System. August 2009.
- [6] ITU-T and ISO/IEC JTC 1: Advanced Video Coding for Generic Audiovisual Services. ITU-T Rec. H.264/AVC and ISO/IEC 14496-10 (including SVC extension). March 2009.
- [7] Advanced Television System Committee: ATSC-Mobile DTV Standard, Part 7 – AVC and SVC Video System Characteristics. October 2009.
- [8] European Broadcasting Union: Draft TS 102 005 V1.4.1: Specification for the use of Video and Audio Coding in DVB services delivered directly over IP protocols. July 2009.
- [9] A. Vetro, C. Christopoulos and H. Sun, "Video Transcoding Architectures and Techniques: an Overview," *IEEE Signal Processing Magazine*. 18--29, 2003.
- [10] H. Shen, S. Xiaoyan, F. Wu, H. Li and S. Li, "Transcoding to FGS Streams from H.264/AVC Hierarchical B-Pictures," *IEEE Int. Conf. Image Processing*, Atlanta, 2006.
- [11] J. De Cock, S. Notebaert, P. Lambert and R. Van de Walle, "Architectures of Fast Transcoding of H.264/AVC to Quality-Scalable SVC Streams," *IEEE Transaction on Multimedia* vol. 11 n.7, pp.1209--1224, 2009.
- [12] R. Sachdeva, S. Johar and E. Piccinelli, "Adding SVC Spatial Scalability to Existing H.264/AVC Video," *8th IEEE/ACIS International Conference on Computer and Information Science*, Shanghai, 2009.
- [13] A. Dziri, A. Diallo, M. Kieffer and P. Duhamel, "P-Picture Based H.264 AVC to H.264 SVC Temporal Transcoding," *International Wireless Communications and Mobile Computing Conference*, 2008.
- [14] H. Schwarz, D. Marpe and T. Wiegand, "Analysis of Hierarchical B pictures and MCTF," *IEEE Int. Conf. ICME and Expo*, Toronto, 2006.

- [15] Joint Video Team JSVM reference software, [http://ip.hhi.de/imagecom\\_G1/savce/downloads/SVC-Reference-Software.htm](http://ip.hhi.de/imagecom_G1/savce/downloads/SVC-Reference-Software.htm)
- [16] Joint Model JM reference software, <http://iphone.hhi.de/suehring/tml/download/>

- [17] G. Sullivan and G. Bjøntegaard, "Recommended Simulation Common Conditions for H.26L Coding Efficiency Experiments on Low-Resolution Progressive-Scan Source Material". ITU-T VCEG, Doc. VCEG-N81, September 2001

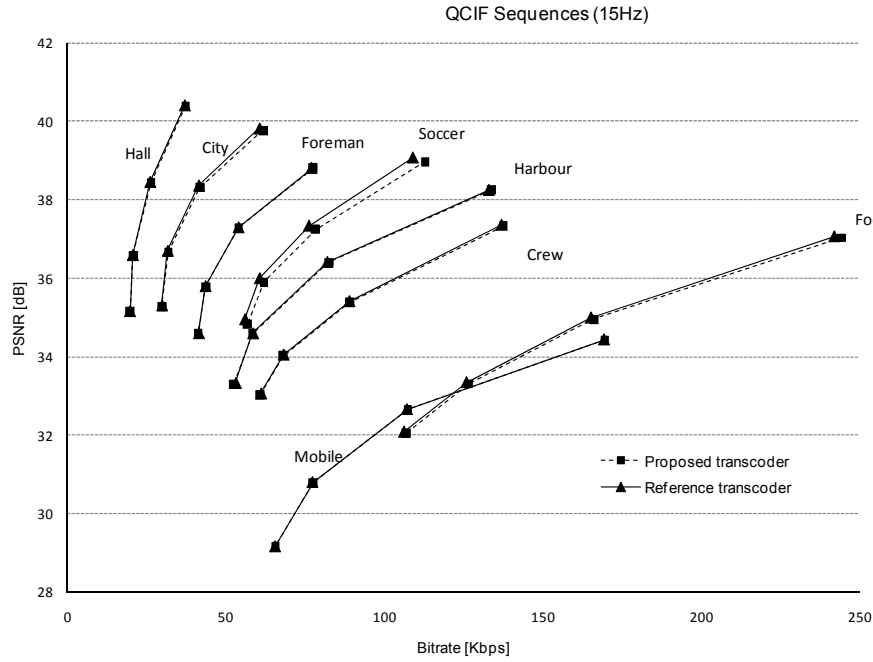


Figure 7. Rate-distortion performance of test sequences in QCIF resolution

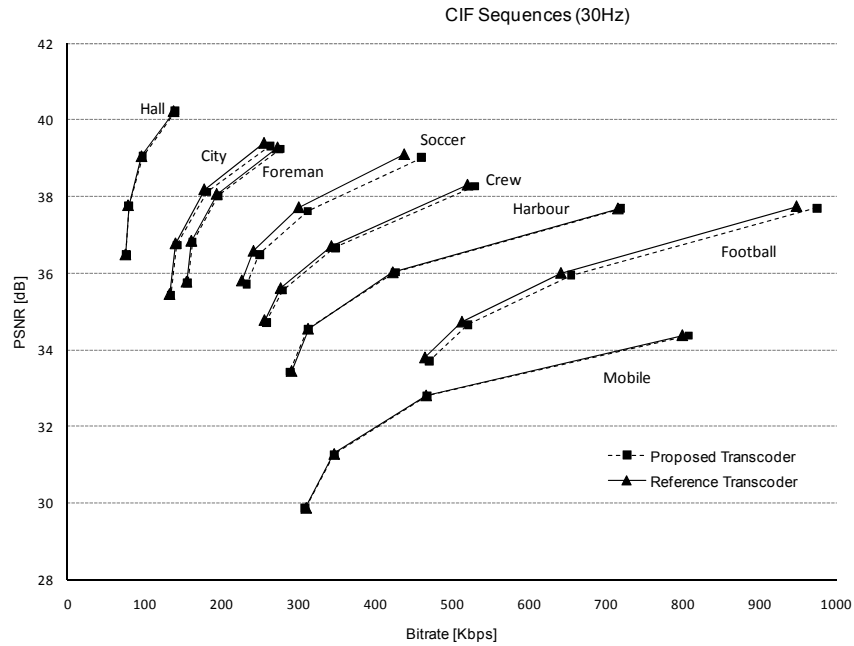


Figure 8. Rate-distortion performance of test sequences in CIF resolution